SOFT MAIL

เทคนิคการสร้าง ไดอะล็อกบ็อกซ์ ในโหมดข้อความ

อธิบายรายละเอียดการทำงานของรูทีนต่าง ๆ ที่ใช้ในการสร้างไดอะล็อกบ็อกซ์

จากตอนที่ผ่าน ๆ มาผมได้เสนอเทคนิคเกี่ยวกับการสร้างเมนูทั้งพูลดาวน์ และ Popup ดังนั้นเพื่อให้มีความสมบูรณ์มากยิ่งขึ้น สำหรับในฉบับนี้ผมจึงได้เสนอเทคนิคการสร้างไดอะล็อกบ็อกซ์ ซึ่งในการเขียนโปรแกรมในการติดต่อหรือแสดงคำสั่งต่าง ๆ นอกจากการใช้เมนูแล้วมักจะมีการแสดงข้อความหรือการตั้งคำถามเพื่อติดต่อกับผู้ใช้โดยใช้ไดอะล็อกบ็อกซ์ชนิดต่าง ๆ เช่น Message box, List box, Input box, Text box, File Selection box เป็นต้น

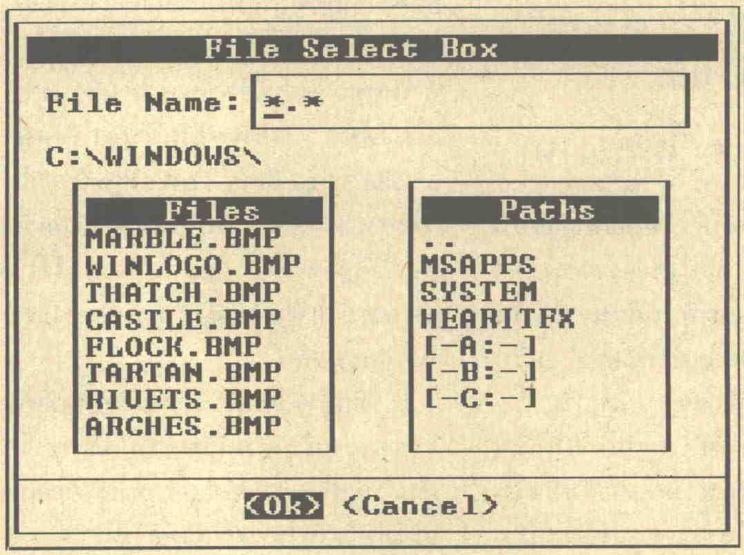
ทำความเข้าใจกันก่อน

และสำหรับเทคนิคการสร้างไดอะล็อกบ็อกซ์ชนิดดังที่กล่าวมาแล้วข้างต้นทั้งหมด ก็ได้ถูกบรรจุเอาไว้ในโปรแกรมตัวอย่าง TDIALOG.BAS แล้วอย่างสมบูรณ์ สำหรับไฟล์ไลบรารีที่ใช้งานร่วมกับโปรแกรมนี้คือ QB.LIB และ ASMTOOLS.LIB ซึ่งมีในหนังสือ QuickBASIC Toolbox จัดจำหน่ายโดยบริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน) และสำหรับหน้าตาของไดอะล็อกชนิดต่าง ๆ ที่ผมได้พัฒนาขึ้นมามีลักษณะดังรูปที่ 1, 2, 3 และ 4

โปรแกรมนี้ยังคงใช้ตัวแปรชนิดโครงสร้าง GCtype เหมือนกับโปรแกรมที่ผ่านมา แต่มีการแก้ไขโดยการเพิ่มและตัดสมาชิกบางตัว ทั้งนี้เพื่อลดขนาดของซอร์สโค้ด ซึ่งสมาชิกแต่ละตัวมีหน้าที่ดังนี้

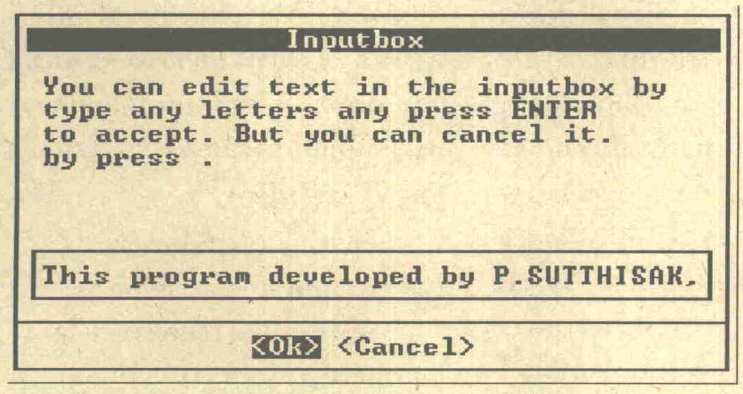
MonoFlag = TRUE หมายถึงโปรแกรมทำงานร่วมกับการ์ดโมโนโครม

File selection box (FileSelectBox% routine)



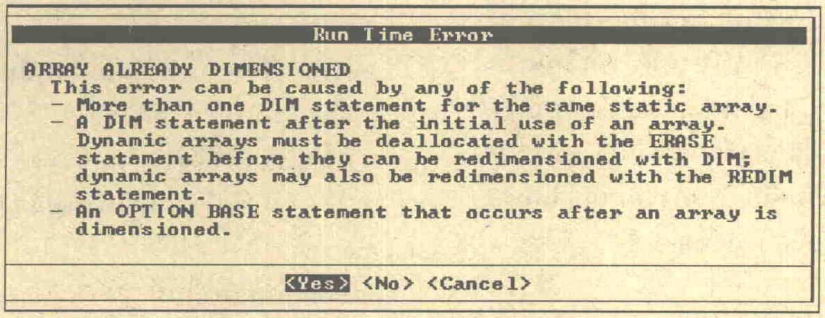
รูปที่ 1

Input box (InputBox% routine)



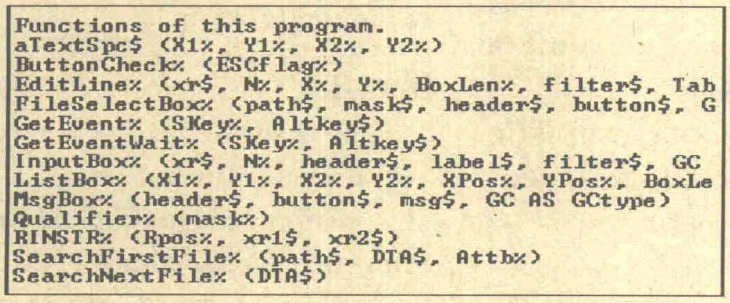
รูปที่ 2

Message box (MsgBox% routine)



รูปที่ 3

Text editor box (TextBox% routine)



รูปที่ 4

หรือการ์ดเฮอร์คิวลิส

MonoFlag = FALSE หมายถึงโปรแกรมทำงานร่วมกับการ์ดซีจีเอ อีจีเอ วีจีเอ หรือ MCGA

MaxCol จำนวนคอลัมน์ที่มากที่สุดสำหรับโหมดการแสดงผลปัจจุบัน

MaxLine จำนวนแถวที่มากที่สุดสำหรับโหมดการแสดงผลปัจจุบัน

WX1 ตำแหน่งคอลัมน์ (X) บนซ้ายของหน้าต่างที่แสดงไดอะล็อกบ็อกซ์

WY1 ตำแหน่งแถว (Y) บนซ้ายของหน้าต่างที่แสดงไดอะล็อกบ็อกซ์

WX2 ตำแหน่งคอลัมน์ (X) ล่างขวาของหน้าต่างที่แสดงไดอะล็อกบ็อกซ์

WY2 ตำแหน่งแถว (Y) ล่างขวาของหน้าต่างที่แสดงไดอะล็อกบ็อกซ์

Caption แอตทริบิวต์สำหรับใช้แสดงผลเฮดเดอร์ของไดอะล็อกบ็อกซ์

Normal แอตทริบิวต์สำหรับใช้แสดงผลแบบปกติ (normal)

Inverse แอตทริบิวต์สำหรับใช้แสดงผลแบบกลับสี (inverse)

ButtonNum จำนวนปุ่มที่สามารถเลือกตอบยอมรับ ยกเลิก หรืออื่น ๆ ของไดอะล็อกบ็อกซ์ ซึ่งปกติจะอยู่ตรงบรรทัดล่างสุดของไดอะล็อกบ็อกซ์

วัตถุประสงค์

สำหรับแอตทริบิวต์ที่มีผลต่อการ์ดและจอภาพแต่ละชนิด ผู้อ่านก็สามารถหาอ่านได้จากฉบับที่ผ่านมา ซึ่งมีโปรแกรมตัวอย่าง ATTBDEMO.BAS ซึ่งแสดงแอตทริบิวต์ของข้อความโดยประโยค COLOR ของ QuickBASIC และ โดยรูทีน aAttbBox ของไฟล์ไลบรารี ASMTOOLS.LIB เนื่องจากโปรแกรมนี้ผมมีเวลาในการพัฒนาที่ค่อนข้างจำกัด และมีโอกาสตรวจสอบไม่บ่อยครั้งนัก ดังนั้นถ้าหากเกิดข้อผิดพลาดขึ้นกับรูทีนใด ๆ ก็ตาม ขอให้ผู้อ่านกรุณาแจ้งผลดังกล่าวกลับมายังกองบรรณาธิการวารสารไมโครคอมพิวเตอร์ บริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน) เพื่อที่จะได้ทำการแก้ไขในโอกาสต่อไป

ในการเขียนโปรแกรมนี้ขึ้นมาผมมีความต้องการให้ผู้อ่านทุกท่านสามารถดึงเอาเฉพาะไดอะล็อกที่ต้องการไปใช้งานโดยง่ายที่สุด ดังนั้นในการสร้างรูทีนต่าง ๆ สำหรับไดอะล็อกบ็อกซ์แต่ละชนิดนั้น ผมพยายามที่จะไม่แยกออกเป็นหลาย ๆ รูทีน ยกเว้นบางส่วนที่หลาย ๆ ไดอะล็อกบ็อกซ์สามารถใช้งาน

ร่วมกันได้เท่านั้นที่จะต้องถูกแยกออกเป็นรูทีนต่าง ๆ ดังนั้นไดอะล็อกบ็อกซ์ที่สร้างโดยเทคนิคนี้จึงไม่ใช้วิธีที่ทำให้ได้ซอร์สโค้ดที่มีขนาดกะทัดรัดที่สุด แต่จะเป็นวิธีที่สามารถทำให้ผู้อ่านสามารถนำไปใช้งานได้ง่ายที่สุด และสำหรับรายละเอียดต่าง ๆ ของแต่ละรูทีนที่มีในโปรแกรมต่อไปนี้มีดังนี้

ชื่อ Alt2ASCII

ชื่อ aTextSpc$

หมายเหตุ มีรายละเอียดในวารสารไมโครคอมพิวเตอร์ ฉบับ 103 ประจำเดือนกุมภาพันธ์ 2537

ButtonCheck%

ชื่อ ButtonCheck%

ประเภท ฟังก์ชัน

หน้าที่ รอคอยการเลือกปุ่มรายการ (Button) และตรวจสอบสถานะของปุ่มรายการ

การประกาศ DECLARE FUNCTION ButtonCheck% (ESCflag%)

พารามิเตอร์

ESCflag% = TRUE ถ้าหากผู้อ่านเลือกปุ่มรายการนี้ (ENTER) ฟังก์ชันก็จะส่งค่ากลับเป็น FALSE

ESCflag% = FALSE ถ้าหากผู้อ่านเลือกปุ่มรายการนี้ (ENTER) ฟังก์ชันก็จะส่งค่ากลับเป็น TRUE

การส่งค่ากลับ ค่าที่ส่งกลับโดยฟังก์ชันนี้มีความหมายดังต่อไปนี้

TRUE หมายถึงผู้อ่านเลือกปุ่มรายการที่มีค่า ESCflag% เป็นเท็จ (FALSE)

FAI.SE หมายถึงผู้อ่านเลือกปุ่มรายการที่มีค่า ESCflag% เป็นจริง (TRUE) หรือ ยกเลิกปุ่มรายการโดยการกด ESC

TABSTOP (9) หมายถึงผู้อ่านเลื่อนไปยังปุ่มรายการถัดไป

หมายเหตุ โดยทั่วไปปุ่มรายการมักจะมีปุ่มใดปุ่มหนึ่งเป็นปุ่มรายการตอบรับ เช่น <Ok>, <Yes> เป็นต้น ดังนั้นค่าของ ESCflag% สำหรับปุ่มเหล่านี้ต้องมีค่าเป็น FALSE ดังนั้นถ้าหากผู้อ่านเลือกปุ่มรายการเหล่านี้โดยการกด Enter ฟังก์ชันนี้ก็จะส่งค่ากลับเป็น TRUE และบางปุ่มเป็นปุ่มของยกเลิก เช่น <Cancel>, <No> เป็นต้น ดังนั้นค่าของ ESCflag% สำหรับปุ่มเหล่านี้ต้องมีค่าเป็น TRUE ดังนั้นถ้าหากผู้อ่านเลือกปุ่มรายการเหล่านี้โดยการกด Enter ฟังก์ชันนี้ก็จะส่งค่ากลับเป็น FALSE (ซึ่งมีค่าเช่นเดียวกับการยกเลิกไดอะล็อกบ็อกซ์โดยการกด ESC)

CountStringLine

ชื่อ CountStringLine

ประเภท โปรแกรมย่อย

หน้าที่ หาจำนวนบรรทัดของข้อความและจำนวนตัวอักษรของบรรทัดที่มีตัวอักษรมากที่สุด

การประกาศ DECLARE SUB CountStringLine (msg$, ChrPerLine%, LineCount%)

พารามิเตอร์

msg$ ข้อความซึ่งแยกออกเป็นบรรทัดด้วยตัวอักษรแอสกี 10

ChrPerLine% จำนวนตัวอักษรของบรรทัดที่มีตัวอักษรมากที่สุด

LineCount% จำนวนบรรทัด

การส่งค่ากลับ ผ่านทางตัวแปร ChrPerLine% และ LineCount%

หมายเหตุ ในการแยกข้อความจากตัวแปร msg$ ออกเป็นบรรทัดนั้นในที่นี้ผมกำหนดให้ตัวอักษรแอสกี 10 เป็นตัวแสดงการจบและขึ้นบรรทัดใหม่ ดังนั้นถ้าหากในข้อความมีตัวอักษรแอสกี 10 ติดต่อกันสองตัวแสดงว่าบรรทัดนั้นเป็นบรรทัดว่าง (Null string)

DispTextInBox

ชื่อ DispTextInBox

ประเภท โปรแกรมย่อย

หน้าที่ แสดงข้อความในไดอะล็อกบ็อกซ์

การประกาศ DECLARE SUB DispTextInBox (msg$, LineCount%, GC AS GCtype)

พารามิเตอร์

msg$ ข้อความซึ่งแยกออกเป็นบรรทัดด้วยตัวอักษรแอสกี 10

LineCount% จำนวนบรรทัด

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ ไม่มี

หมายเหตุ ในการแสดงข้อความภายในรูทีนนี้กระทำโดยรูทีน aOutText ดังนั้นจึงไม่มีการเปลี่ยนแปลงแอตทริบิวต์ของที่เป็นอยู่ในปัจจุบัน

DrawBox

ชื่อ DrawBox

ประเภท โปรแกรมย่อย

หน้าที่ แสดงกรอบสี่เหลี่ยมและกรอบสำหรับไดอะล็อกบ็อกซ์

การประกาศ DECLARE SUB DrawBox (X1%, Y1%, X2%, Y2% DialogFlag%)

พารามิเตอร์

X1%, Y1% ตำแหน่งคอลัมน์และแถวบนซ้ายของกรอบ

X2%, Y2% ตำแหน่งคอลัมน์และแถวล่างขวาของกรอบ

DialogFlag% = FALSE แสดงกรอบสี่เหลี่ยม

DialogFlag% = TRUE แสดงกรอบสี่เหลี่ยมสำหรับไดอะล็อกบ็อกซ์

การส่งค่ากลับ ไม่มี

หมายเหตุ สำหรับรูทีนนี้เป็นรูทีนที่ได้รับการแก้ไขจากโปรแกรมที่ผ่านมาโดยการเพิ่มตัวแปร DialogFlag% เพื่อให้สามารถแสดงกรอบสำหรับไดอะล็อกได้ด้วย

DrawButton

ชื่อ DrawButton

ประเภท โปรแกรมย่อย

หน้าที่ แสดงปุ่มรายการ (button) ของไดอะล็อกบ็อกซ์

การประกาศ DECLARE SUB DrawButton (button$, Butflag%, GC AS GCtype)

พารามิเตอร์

button$ ปุ่มรายการทั้งหมด โดยที่จะแยกแต่ละปุ่มรายการด้วยตัวอักษร "I" (แอสกี 58)

Butflag% ปุ่มรายการที่ต้องการกำหนดให้เป็นปุ่มที่กำลังทำงาน (Active) หรือปุ่มรายการดีฟอลต์

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ ไม่มี

หมายเหตุ สำหรับตัวอย่างการกำหนดปุ่มรายการลงในตัวแปร button$ เช่น

```
button$ = "Ok|Cancel|Help"
Butflag% = 1
```

ซึ่งรูทีนนี้ก็จะแยกออกเป็น 3 ปุ่มรายการดังนี้

```
<Ok> <Cancel> <Help>
```

และกำหนดให้ปุ่มรายการ <Ok> มีสถานะแอคตีฟ

DrawDialog

ชื่อ DrawDialog

ประเภท โปรแกรมย่อย

หน้าที่ แสดงรูปร่างของไดอะล็อกบ็อกซ์

การประกาศ DECLARE SUB DrawDialog (header$, button$, GC AS GCtype)

พารามิเตอร์

header$ ข้อความที่แสดงในส่วนหัว (caption) ของไดอะล็อกบ็อกซ์

button$ ปุ่มรายการทั้งหมด โดยที่จะแยกแต่ละปุ่มรายการด้วยตัวอักษร "I" (แอสกี 58)

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ ไม่มี

หมายเหตุ ไม่มี

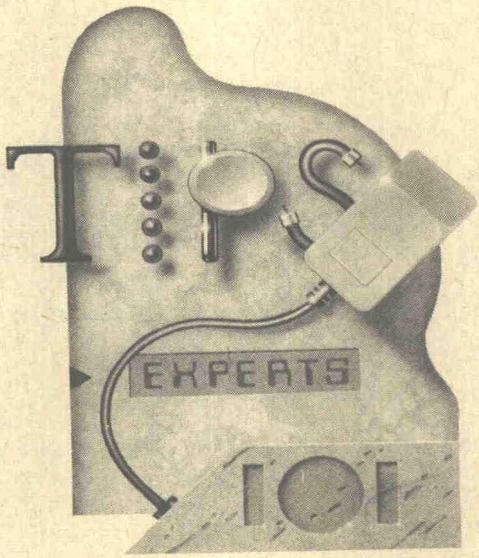
EditLine%

ชื่อ EditLine%

ประเภท ฟังก์ชัน

หน้าที่ แก้ไขข้อความในหนึ่งบรรทัด (One line text editor)

การประกาศ DECLARE FUNCTION EditLine% (xr$, N%, X%, Y%, BoxLen%, filter$, TabFlag%, GC AS GCtype)



พารามิเตอร์

xr$ ข้อความที่ต้องการแก้ไข

N% จำนวนตัวอักษรที่มากที่สุดที่สามารถแก้ไข

Y% ตำแหน่งแถว (Y) บนซ้ายของไดอะล็อกบ็อกซ์

BoxLen% ความกว้างของขอบเขตที่แสดงผลข้อความที่กำลังแก้ไข

filter$ <> "" หมายถึงรายการตัวอักษรที่สามารถแก้ไขได้ (ตัวอักษรที่ยอมรับ)

filter$ = "" หมายถึงสามารถแก้ไขตัวอักษรได้ทุก ๆ ตัวอักษร

TabFlag% = TRUE ถ้าหากผู้อ่านกดคีย์ TABSTOP (9) ฟังก์ชันก็จะส่งค่ากลับเป็น 1 ซึ่งค่าดังกล่าวนี้มีไว้เพื่อประโยชน์ในการนำไปใช้ในการสร้างไดอะล็อกบ็อกซ์ และสามารถเลื่อนไปมาระหว่างแต่ละรายการภายในโดยการกดคีย์ TABSTOP

TabFlag% = FALSE การกดคีย์ TABSTOP จะไม่ถูกนำมาสนใจ

การส่งค่ากลับ ค่าที่ส่งกลับโดยฟังก์ชันนี้มีความหมายดังต่อไปนี้

TRUE หมายถึงผู้อ่านยอมรับข้อความที่ถูกแก้ไขโดยการกดคีย์ Enter และข้อความที่ถูกแก้ไขก็ถูกส่งกลับโดยผ่านทางตัวแปร xr$

FALSE หมายถึงผู้อ่านยกเลิกการแก้ไขข้อความโดยการกดคีย์ ESC

1 หมายถึงผู้อ่านเลื่อนไปยังปุ่มรายการถัดไป และข้อความที่ถูกแก้ไขก็ถูกส่งกลับโดยผ่านทางตัวแปร xr$ ถ้าหาก TabFlag% = TRUE

หมายเหตุ ข้อความที่ถูกแก้ไขและถูกส่งกลับผ่านทางตัวแปร xr$ จะเป็น

ข้อความที่ถูกกำจัดตัวอักษรช่องว่าง (แอสกี 32) ออกจนหมดโดยฟังก์ชัน RTRIM$ ของ QuickBASIC

คีย์ที่ใช้ทำงานร่วมกับฟังก์ชัน

EditLine%

สำหรับคีย์ต่าง ๆ ที่สามารถทำงานได้มีดังนี้

BACKSPACE ลบตัวอักษรทางซ้ายมือของเคอร์เซอร์ 1 ตัวอักษร

TABSTOP ถ้าหาก TabFlag% =TRUE เลื่อนไปยังรายการต่อไป

ENTER ยอมรับการแก้ไข และฟังก์ชันส่งค่ากลับเป็น TRUE

ESC ยกเลิกการแก้ไข และฟังก์ชันส่งค่ากลับเป็น FALSE

HOME เลื่อนเคอร์เซอร์ไปยังอักษรตัวแรกของข้อความ

LEFT เลื่อนเคอร์เซอร์ไปยังตัวอักษรทางซ้าย 1 ตัวอักษร

RIGHT เลื่อนเคอร์เซอร์ไปยังตัวอักษรทางขวา 1 ตัวอักษร

END เลื่อนเคอร์เซอร์ไปยังตัวอักษรตัวสุดท้ายของข้อความ ถ้าหากข้อความยาวเท่ากับ N% หรือหลังตัวอักษรตัวสุดท้ายของข้อความ ถ้าหากข้อความยาวน้อยกว่า N%

INSERT เปลี่ยนไปมาระหว่างโหมดพิมพ์ทับและโหมดพิมพ์แทรก

DEL ลบตัวอักษรที่ตำแหน่งเคอร์เซอร์ 1 ตัวอักษรและดึงข้อความทางซ้ายมือของเคอร์เซอร์เข้ามา

CTRL+LEFT เลื่อนเคอร์เซอร์ไปยังคำทางซ้ายมือ 1 คำ

CTRL+RIGHT เลื่อนเคอร์เซอร์ไปยังคำทางขวามือ 1 คำ

FileSelectBox%

ชื่อ FileSelectBox%

ประเภท ฟังก์ชัน

หน้าที่ แสดงไดอะล็อกบ็อกซ์สำหรับเลือกไฟล์ (File selection box)

การประกาศ DECLARE FUNCTION FileSelectBox% (path$, mask$, header$, button$, GC AS GCtype)

พารามิเตอร์

path$ ไดเรคทอรีสำหรับค้นหาไฟล์

mask$ ชื่อไฟล์สำหรับค้นหา ซึ่งสามารถกำหนดนามสกุลได้ด้วย

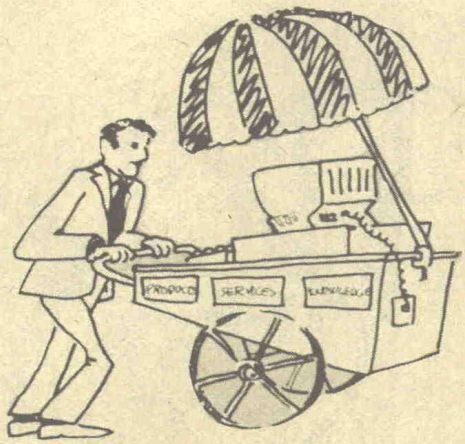
header$ ข้อความที่แสดงในส่วนหัว (caption) ของไดอะล็อกบ็อกซ์

button$ ปุ่มรายการทั้งหมด โดยที่จะแยกแต่ละปุ่มรายการด้วยตัวอักษร "I" (แอสกี 58)

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ

TRUE ผู้อ่านยอมรับไฟล์ที่เลือกและไดเรคทอรี และชื่อไฟล์ที่ถูกเลือกจะถูกส่งกลับผ่านมาทางตัวแปร path$ และ mask$



FALSE ผู้อ่านยกเลิกการเลือกไฟล์

หมายเหตุ สำหรับผู้ที่เคยใช้งานโปรแกรม BEQB เวอร์ชัน 1.00 ก็คงจะคุ้นเคยกับไดอะล็อกบ็อกซ์นี้ดีแล้ว แต่สำหรับผู้ที่ไม่เคยใช้งานก็ไม่ยาก ซึ่งผู้อ่านสามารถเลื่อนไปมาระหว่างรายการได้โดยการกดคีย์ TABSTOP และยอมรับการเลือกโดยการกดคีย์ Enter หรือเลือกปุ่มรายการ Ok หรือยกเลิกการเลือกโดยการกดคีย์ ESC หรือเลือกปุ่มรายการ Cancel

GetDTAaddress

ชื่อ GetDTAaddress

หน้าที่ ค้นหาตำแหน่งของ DTA (Disk Transfer Area)

หมายเหตุ ดูรายละเอียดจากโปรแกรม FILES.BAS ของหนังสือ QuickBASIC ToolBox

ชื่อ GetEvent%

ชื่อ GetEventWait%

ชื่อ GetVideo

หมายเหตุ มีรายละเอียดในวารสารไมโครคอมพิวเตอร์ ฉบับ 103 ประจำเดือนกุมภาพันธ์ 2537

InputBox%

ชื่อ InputBox%

ประเภท ฟังก์ชัน

หน้าที่ ไดอะล็อกบ็อกซ์ชนิดกรอกข้อความ (input box)

การประกาศ DECLARE FUNCTION InputBox% (xr$, N%, header$, label$, filter$, GC AS GCtype)

พารามิเตอร์

xr$ ข้อความที่ต้องการแก้ไข

N% จำนวนตัวอักษรที่มากที่สุดที่สามารถแก้ไข

header$ ข้อความที่แสดงในส่วนหัว (caption) ของไดอะล็อกบ็อกซ์

label$ ข้อความที่แสดงเพื่อสื่อความหมายหรือคำถามกับผู้อ่าน

filter$ <> " " หมายถึงรายการตัวอักษรที่สามารถแก้ไขได้ (ตัวอักษรที่ยอมรับ)

filter$ = " " หมายถึงสามารถแก้ไขตัวอักษรได้ทุก ๆ ตัวอักษร

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ

= TRUE ผู้อ่านยอมรับการแก้ไขข้อความ และข้อความที่ถูกแก้ไขก็จะถูกส่งกลับโดยผ่านมายังตัวแปร xr$ ซึ่งข้อความที่ถูกส่งกลับนี้จะถูกจำกัดตัวอักษรช่องว่างทางขวาของข้อความออกทั้งหมด

= FALSE ผู้อ่านยกเลิกการแก้ไข

หมายเหตุ สำหรับการเลื่อนไปมาระหว่างแต่ละรายการของไดอะล็อกบ็อกซ์ สามารถทำได้โดยการกดคีย์ TABSTOP

ชื่อ KeyPressed

หมายเหตุ มีรายละเอียดในวารสารไมโครคอมพิวเตอร์ ฉบับ 103 ประจำเดือนกุมภาพันธ์ 2537

ListBox%

ชื่อ ListBox%

ประเภท ฟังก์ชัน

หน้าที่ แสดงไดอะล็อกบ็อกซ์ชนิด List Box

การประกาศ DECLARE FUNCTION ListBox% (X1%, Y1%, X2%, Y2%, XPos%, YPos%, BoxLen%, item$( ), GC AS GCtype)

พารามิเตอร์

X1%, Y1% ตำแหน่งคอลัมน์และแถวบนซ้ายของพื้นที่สำหรับ List box

X2%, Y2% ตำแหน่งคอลัมน์และแถวล่างขวาของพื้นที่สำหรับ List box

XPos%, YPos% ตำแหน่งคอลัมน์และแถว สำหรับแสดงรายการที่ถูกเลือก

BoxLen% ความกว้างของขอบเขตที่แสดงผลข้อความที่ถูกเลือก

item$( ) รายการทั้งหมดที่ถูกแสดงโดย List box

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ

TRUE ถ้าหากผู้อ่านกดคีย์ TABSTOP ซึ่งหมายถึงผู้อ่านต้องการที่จะเลื่อนไปยังรายการถัดไปของไดอะล็อกบ็อกซ์

FALSE ถ้าหากผู้อ่านยกเลิกการเลือกรายการ

>= 1 ผู้อ่านยอมรับการเลือกรายการ และค่าดังกล่าวก็เป็นรายการที่ถูกผู้อ่านเลือก

หมายเหตุ สำหรับไดอะล็อกบ็อกซ์ Listbox แท้ที่จริงก็คือ Popup menu นั่นเอง แต่ผมได้ทำการแก้ไขเพื่อให้สามารถนำมาใช้ร่วมกับ File select box ได้ด้วย ดังนั้นถ้าหากผู้อ่านต้องการให้มันทำหน้าที่เป็น Popup menu ก็ให้กำหนดให้ XPos% = 0 และไม่ต้องสนใจค่าของ YPos% และ BoxLen% และเรียกใช้รูทีนนี้โดยมีลำดับดังรูปที่ 5

```
X1% = GC.WX1+1 : Y1% = GC.WY1+1
X2% = GC.WX2-1: Y2% = GC.WY2-1
scrnbuf$ = aTextSpc$ (GC.WX1, GC.WY1, GC.WX2, GC.WY2)
aKeepBox GC.WX1, GC.WY1, GC.WX2, GC.WY2, scrnBuf$
DrawBox GC.WX1, GC.WY1, GC.WX2, GC.WY2
Selected% = ListBox% (X1%, Y1%, X2%, Y2%, 0, 0, 0, items$( ) GC)
aRestoreBox GC.WX1, GC.WY1, GC.WX2, GC.WY2, scrnBuf$
```

รูปที่ 5

MsgBox%

ชื่อ MsgBox%

ประเภท ฟังก์ชัน

หน้าที่ แสดงไดอะล็อกบ็อกซ์ชนิดแสดงข้อความ (Message box)

การประกาศ DECLARE FUNCTION MsgBox% (header$, button$, msg$, GC AS GCtype)

พารามิเตอร์

header$ ข้อความที่แสดงในส่วนหัว (caption) ของไดอะล็อกบ็อกซ์

button$ ปุ่มรายการทั้งหมด โดยที่จะแยกแต่ละปุ่มรายการด้วยตัวอักษร "I" (แอสกี 58)

msg$ ข้อความที่ต้องการแสดงในไดอะล็อก ซึ่งแต่ละบรรทัดของข้อความจะถูกแยกออกจากกันโดยเครื่องหมาย "I"

GC ตัวแปรชนิดโครงสร้าง GCtype

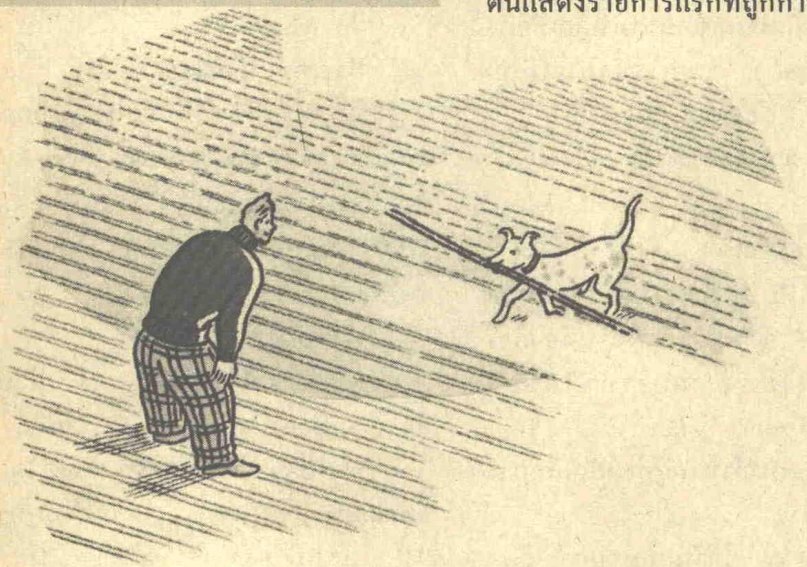
การส่งค่ากลับ

FALSE ถ้าหากผู้อ่านยกเลิกการเลือก

>= 1 ผู้อ่านเลือกปุ่มรายการหมายเลขนั้น ๆ

หมายเหตุ สำหรับจำนวนตัวอักษรในแต่ละบรรทัดที่มากที่สุดที่สามารถแสดงผลได้นั้นจะต้องน้อยกว่าความกว้างของจอภาพอยู่ 4 ตัวอักษร และเช่นเดียวกันจำนวนบรรทัดของข้อความที่มากที่สุดที่สามารถแสดงผลได้จะต้องน้อยกว่าจำนวนบรรทัดของจอภาพอยู่ 4 บรรทัด ดังนั้นถ้าหากผู้อ่านต้องการที่จะแก้ไขค่าดังกล่าวข้างต้นก็ขอให้แก้ไขจาก 2 บรรทัดข้างล่างนี้ซึ่งอยู่ในรูทีนนี้

```
    IF ChrPerLine% > (GC.MaxCol-4)
THEN ChrPerLine% = GC.MaxCol-4
    IF LineCount% > (GC.MaxLine-4)
THEN LineCount% = GC.MaxLine-4
```



PrintMenu

ชื่อ PrintMenu

ประเภท โปรแกรมย่อย

หน้าที่ แสดงรายการในกรอบสี่เหลี่ยมที่กำหนด

การประกาศ DECLARE SUB PrintMenu (X1%, Y1%, First%, MaxItem%, ItemLen%, item$( ), GC AS GCtype)

พารามิเตอร์

X1%, Y1% ตำแหน่งคอลัมน์และแถวบนซ้ายของพื้นที่สำหรับ List box

First% หมายเลขกำหนดรายการแรกที่จะแสดงที่บรรทัด Y1%

MaxItem% จำนวนรายการที่สามารถแสดงได้มากที่สุด

ItemLen% ความยาวของแต่ละรายการที่สามารถแสดงได้มากที่สุด

item$( ) รายการทั้งหมดที่ถูกแสดงโดย List box

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ ไม่มี

หมายเหตุ ในการแสดงผลจะเริ่มต้นแสดงรายการแรกที่ถูกกำหนดในตัวแปร First% ที่ตำแหน่ง X1%, Y1%

ชื่อ Qualifier%

หมายเหตุ มีรายละเอียดในวารสารไมโครคอมพิวเตอร์ ฉบับ 103 ประจำเดือนกุมภาพันธ์ 2537

RINSTR%

ชื่อ RINSTR%

ประเภท ฟังก์ชัน

หน้าที่ ค้นหากลุ่มตัวอักษรจากข้อความแบบย้อนกลับ

การประกาศ DECLARE FUNCTION RINSTR% (Rpos%, xr1$, xr2$)

พารามิเตอร์

Rpos% ตำแหน่งบน xr1$ สำหรับเริ่มต้นการค้นหา

xr1$ ข้อความที่ถูกค้นหา

xr2$ กลุ่มของตัวอักษรที่ต้องการค้นหา

การส่งค่ากลับ

0 ถ้าหากไม่สามารถพบ x2$ ใน x1$

>= 1 ถ้าหากค้นพบตัวอักษรตัวใดตัวหนึ่งของ x2$ ใน x1$

หมายเหตุ สำหรับฟังก์ชันนี้จะมีวิธีการค้นหาที่ต่างไปจากคำสั่ง INSTR ของ QuickBASIC ซึ่งมีการทำงานดังนี้

xr1$ หมายถึงข้อความที่จะถูกค้นหา

และ xr1$ หมายถึงกลุ่มของตัวอักษรที่ต้องการค้นหา

ถ้าหากมีการค้นพบตัวอักษรตัวใดตัวหนึ่งของ xr2$ ใน xr1$ ฟังก์ชันนี้ก็จะหยุดการทำงานทันที และส่งกลับ

ตำแหน่งที่ค้นพบตัวอักษรดังกล่าวทันที โดยในการค้นหาจะเริ่มต้นจากตำแหน่งที่ถูกกำหนดโดยตัวแปร Rpos% จนถึงอักษรตัวแรกของ xr1$

ชื่อ ScrollBox

หมายเหตุ มีรายละเอียดในวารสารไมโครคอมพิวเตอร์ ฉบับ 103 ประจำเดือนกุมภาพันธ์ 2537

SearchFirstFile%

ชื่อ SearchFirstFile%

หน้าที่ ค้นหาไฟล์ครั้งแรก

หมายเหตุ ถ้าหากฟังก์ชันสามารถค้นหาไฟล์แรกพบ ข้อมูลเกี่ยวกับไฟล์ดังกล่าวก็จะถูกส่งกลับผ่านทางตัวแปร DTA$ ส่วนรายละเอียดอื่น ๆ ผู้อ่านสามารถหาอ่านเพิ่มเติมได้จากโปรแกรม FILES.BAS ของหนังสือ QuickBASIC Toolbox

SearchNextFile%

ชื่อ SearchNextFile%

หน้าที่ ค้นหาไฟล์ครั้งต่อไป

หมายเหตุ สำหรับฟังก์ชันนี้จะต้องถูกเรียกหลังฟังก์ชัน SearchFirstFile% เท่านั้น ส่วนรายละเอียดอื่น ๆ ผู้อ่านสามารถหาอ่านเพิ่มเติมได้จากโปรแกรม FILES.BAS ของหนังสือ QuickBASIC Toolbox

SetDTAaddress

ชื่อ SetDTAaddress

หน้าที่ กำหนดตำแหน่งของ DTA (Disk Transfer Area)

หมายเหตุ ดูรายละเอียดจากโปรแกรม FILES.BAS ของหนังสือ QuickBASIC ToolBox

ชื่อ SetGCtype

หมายเหตุ มีรายละเอียดในวารสารไมโครคอมพิวเตอร์ ฉบับ 103 ประจำเดือนกุมภาพันธ์ 2537

ShowPageOfTextBox

ชื่อ ShowPageOfTextBox

ประเภท โปรแกรมย่อย

หน้าที่ แสดงข้อความในไดอะล็อกบ็อกซ์ชนิด Text box

การประกาศ DECLARE SUB ShowPageOfTextBox (text$( ), FirstPos%, FirstLine%, FromRow%, ToRow%, GC AS GCtype)

พารามิเตอร์

text$( ) อะเรย์สำหรับเก็บข้อความแต่ละบรรทัดของไดอะล็อกบ็อกซ์ชนิด Text box

FirstPos% ตำแหน่งแรกของข้อความที่จะถูกแสดงตรงซ้ายสุดของ Text box

FirstLine% บรรทัดแรกของข้อความที่จะถูกแสดงตรงตำแหน่ง FromRow%

FromRow% แถวแรกที่จะเริ่มต้นแสดงผล

ToRow% แถวสุดท้ายสำหรับการแสดงผล

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ ไม่มี

หมายเหตุ รูทีนนี้ถูกเรียกใช้โดยฟังก์ชัน TextBox% เท่านั้น

TextBox%

ชื่อ TextBox%

ประเภท ฟังก์ชัน

หน้าที่ ไดอะล็อกบ็อกซ์ชนิด Text box

การประกาศ DECLARE FUNCTION TextBox% (msg$, GC AS GCtype)

พารามิเตอร์

msg$ ข้อความที่ต้องการแก้ไข ซึ่งแต่ละบรรทัดจะถูกแยกออกจากกันด้วยตัวอักษรแอสกี 10

GC ตัวแปรชนิดโครงสร้าง GCtype

การส่งค่ากลับ

TRUE ถ้าหากผู้อ่านยอมรับการแก้ไขข้อความ และข้อความที่ถูกแก้ไขจะถูกส่งกลับผ่านมาทางตัวแปร msg$ โดยที่แต่ละบรรทัดของข้อความจะถูกแยกออกจากกันด้วยตัวอักษรแอสกี 10 เช่นเดิม

FALSE ผู้ใช้ยกเลิกการแก้ไขโดยการกดคีย์ ESC

หมายเหตุ ฟังก์ชันนี้แท้ที่จริงก็คือ Text editor นั่นเอง เพียงแต่มีข้อดีกว่าคือสามารถกำหนดขอบเขตสี่เหลี่ยมที่จะให้แสดงผลได้ โดยที่ขอบเขตของ

ไดอะล็อกบ็อกซ์จะถูกกำหนดไว้ในตัวแปร GC.WX1, GC.WY1, GC.WX2 และ GC.WY2 และตัวมันแสดงผลเป็นแบบ popup window ดังนั้นผู้อ่านจึงสามารถแสดง Text editor ได้พร้อม ๆ กันทีละหลาย ๆ ตัว แต่สำหรับข้อจำกัดของฟังก์ชันนี้มีดังนี้

1. จำนวนตัวอักษรต่อบรรทัดที่มากที่สุดเท่ากับ 255 ตัวอักษร

2. จำนวนบรรทัดที่มากที่สุดเท่ากับ 100 บรรทัด แต่ผู้อ่านสามารถเพิ่มจำนวนบรรทัดที่มากที่สุดได้โดยการแก้ไขข้อความในบรรทัดดังนี้

```
ArrNum% = 100
```

คีย์ที่ใช้งานร่วมกับฟังก์ชัน

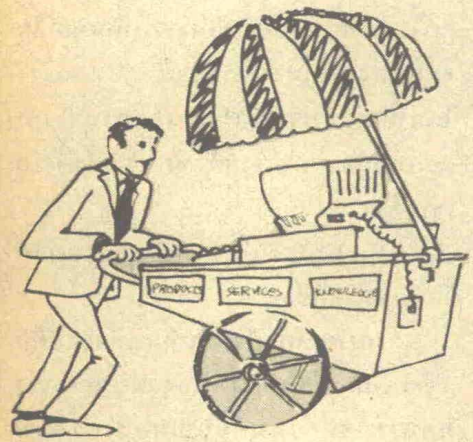
TextBox%

สำหรับคีย์ที่สามารถใช้งานร่วมกับฟังก์ชันนี้ได้มีดังนี้

BACKSPACE ลบตัวอักษรทางซ้ายมือของเคอร์เซอร์ 1 ตัวอักษร

TABSTOP ยอมรับการแก้ไขและฟังก์ชันส่งค่ากลับเป็น TRUE

ENTER ขึ้นบรรทัดใหม่ โดยถ้าหากตำแหน่งของเคอร์เซอร์มีข้อความต่อไปอีก ตัวอักษรตั้งแต่ตำแหน่งเคอร์เซอร์ก็จะถูกตัดเป็นบรรทัดใหม่ด้วย



ESC ยกเลิกการแก้ไขและฟังก์ชันส่งค่ากลับเป็น FALSE

PGUP เลื่อนข้อความขึ้นข้างบนทีละ 1 จอภาพ

PGDN เลื่อนข้อความลงข้างล่างทีละ 1 จอภาพ

HOME เลื่อนเคอร์เซอร์ไปยังอักษรตัวแรกของบรรทัด

END เลื่อนเคอร์เซอร์ไปยังตัวอักษรตัวสุดท้ายของบรรทัดถ้าหากข้อความยาวเท่ากับ 255 หรือหลังตัวอักษรตัวสุดท้ายของข้อความถ้าหากข้อความยาวน้อยกว่า 255

LEFT เลื่อนเคอร์เซอร์ไปยังตัวอักษรทางซ้าย 1 ตัวอักษร

RIGHT เลื่อนเคอร์เซอร์ไปยังตัวอักษรทางขวา 1 ตัวอักษร

UP เลื่อนเคอร์เซอร์ขึ้นข้างบนทีละ 1 บรรทัด

DOWN เลื่อนเคอร์เซอร์ลงข้างล่างทีละ 1 บรรทัด

INSERT เปลี่ยนไปมาระหว่างโหมดพิมพ์ทับและโหมดพิมพ์แทรก

DEL ลบตัวอักษรที่ตำแหน่งเคอร์เซอร์ 1 ตัวอักษร และดึงข้อความทางซ้ายมือของเคอร์เซอร์เข้ามา แต่ถ้าหากเคอร์เซอร์อยู่ที่ตำแหน่งตัวอักษรตัวสุดท้ายของบรรทัด บรรทัดถัดไปก็จะถูกดึงขึ้นมารวมกับบรรทัดปัจจุบันของเคอร์เซอร์

CTRL+LEFT เลื่อนเคอร์เซอร์ไปยังคำทางซ้ายมือ 1 คำ

CTRL+RIGHT เลื่อนเคอร์เซอร์ไปยังคำทางขวามือ 1 คำ

CTRL+PGUP เลื่อนเคอร์เซอร์ไปทางขวามือทีละ 1 จอภาพ

CTRL+PGDN เลื่อนเคอร์เซอร์ไปทางซ้ายมือทีละ 1 จอภาพ

CTRL+HOME เลื่อนไปยังแถวแรกของข้อความ และกำหนดให้เคอร์เซอร์อยู่ที่ตำแหน่งคอลัมน์แรกของบรรทัดแรก

CTRL+END เลื่อนไปยังแถวสุดท้ายของข้อความ และกำหนดให้เคอร์เซอร์อยู่ที่ตำแหน่งคอลัมน์แรกของบรรทัดสุดท้าย

Ztrim$

ชื่อ Ztrim$

ประเภท ฟังก์ชัน

หน้าที่ กำจัดตัวอักษรแอสกี 0 ทั้งหมดออกจากทางขวาของสตริง

การประกาศ DECLARE FUNCTION Ztrim$ (xr$)

พารามิเตอร์ xr$ สตริงที่กำหนด

การส่งค่ากลับ Ztrim$ สตริงที่ถูกกำจัดตัวอักษรแอสกี 0 ทั้งหมดออกจากทางขวาของสตริง

หมายเหตุ ฟังก์ชันนี้ทำงานคล้ายกับฟังก์ชัน RTRIM$ ของ QuickBASIC เพียงแต่ฟังก์ชัน RTRIM$ จะกำจัดตัวอักษรช่องว่างแอสกี 32 แต่ฟังก์ชัน Ztrim$ จะกำจัด ASCII 0 เท่านั้น

สำหรับลิสติงที่ใช้ในการสร้างไดอะล็อกบ็อกซ์คือ TDIALOG.BAS แต่ในฉบับนี้ทางวารสารไม่สามารถที่จะนำลิสติงมาลงให้ท่านผู้อ่านศึกษาได้ เนื่องจากลิสติง TDIALOG.BAS นี้มีความยาวมากเกินไป ดังนั้นขอให้ท่านผู้อ่านติดตามลิสติงนี้ได้จากซอฟต์เมล์ของฉบับนี้นะครับ